



- (51) International Patent Classification: Not classified
- (21) International Application Number: PCT/US2012/038814
- (22) International Filing Date: 21 May 2012 (21.05.2012)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data: 61/487,944 19 May 2011 (19.05.2011) US
- (72) Inventors; and
- (71) Applicants : **BHANU, Bir** [US/US]; c/o The Regents of the University of California, Chemical Science Building, Room 136, 501 Big Springs Road, Riverside, California 92521 (US). **KAFAL, Mehran** [US/US]; c/o The Regents of the University of California, Chemical Science Building, Room 136, 501 Big Springs Road, Riverside, California 92521 (US).
- (74) Agents: **NUZUM, Kirk M.** et al.; Buchanan Ingersoll & Rooney PC, P. O. Box 1404, Alexandria, Virginia 22313-1404 (US).

- (81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— without international search report and to be republished upon receipt of that report (Rule 48.2(g))

(54) Title: DYNAMIC BAYESIAN NETWORKS FOR VEHICLE CLASSIFICATION IN VIDEO

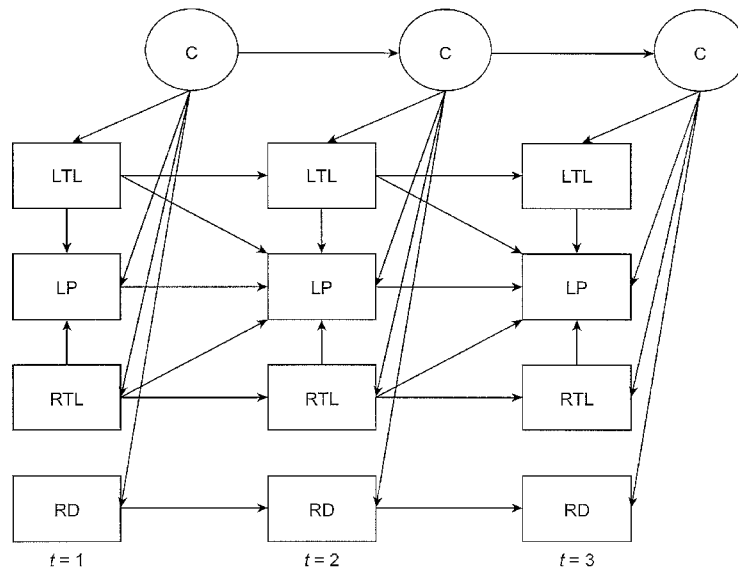


FIG. 7

(57) Abstract: A system and method for classification of passenger vehicles and measuring their properties, and more particularly to a stochastic multi-class vehicle classification system, which classifies a vehicle (given its direct rear-side view) into one of four classes Sedan, Pickup truck, SUV/Minivan, and unknown, and wherein a feature pool of tail light and vehicle dimensions is extracted which feeds a feature selection algorithm to define a low-dimensional feature vector, and the feature vector is then processed by a Hybrid Dynamic Bayesian Network (HDBN) to classify each vehicle.

WO 2012/159109 A2

**DYNAMIC BAYESIAN NETWORKS FOR VEHICLE CLASSIFICATION
IN VIDEO**

CROSS REFERENCE TO RELATED APPLICATIONS

5 [0001] This application claims priority under 35 U.S.C. §119 to U.S. Provisional Application No. 61/487,944 entitled DYNAMIC BAYESIAN NETWORKS FOR VEHICLE CLASSIFICATION IN VIDEO, filed May 19, 2011, the entire content of which is hereby incorporated by reference.

GOVERNMENT INTEREST

10 [0002] This invention was made with Government support under Grant No. IIS0905671 awarded by National Science Foundation. The Government has certain rights in this invention.

FIELD OF THE INVENTION

15 [0003] The present invention relates to a system and method for classification of passenger vehicles and measuring their properties, and more particularly to a stochastic multi-class vehicle classification system, which classifies a vehicle (given its direct rear-side view) into one of four classes Sedan, Pickup truck, SUV/Minivan, and unknown, and wherein a feature pool of tail light and vehicle dimensions is
20 extracted, which feeds a feature selection algorithm to define a low-dimensional feature vector, and the feature vector is then processed by a Hybrid Dynamic Bayesian Network (HDBN) to classify each vehicle.

BACKGROUND OF THE INVENTION

25 [0004] Over the past few years vehicle classification has been widely studied as part of the broader vehicle recognition research area. A vehicle classification system is essential for effective transportation systems (e.g., traffic management and toll systems), parking optimization, law enforcement, autonomous navigation, etc. A common approach utilizes vision-based methods to detect and classify a vehicle in
30 still images and video streams. A human being may be capable of identifying the

class of a vehicle with a quick glance at the digital data (image, video) but accomplishing that with a computer is not as straight forward. Several problems such as occlusion, tracking a moving object, shadows, rotation, lack of color invariance, and many more must be carefully considered in order to design an effective and robust automatic vehicle classification system. Much research has been conducted for object classification, but vehicle classification has shown to have its own specific problems, which motivates research in this area.

[0005] Not much has been done on vehicle classification from the rear view. For the side view, appearance based methods especially edge-based methods have been widely used for vehicle classification. These approaches utilize various methods such as weighted edge matching, Gabor features, edge models, shape based classifiers, part based modeling, and edge point groups. Model-based approaches that use additional prior shape information have also been investigated in 2D (two-dimensions) and more recently in 3D (three-dimensions).

[0006] Vehicle make and model classification from the frontal view has also been investigated (i.e., high resolution, close up frontal view images, neural network classifier), and also (SIFT features). For the rear view, Dlagnekov extends a new license plate recognition system to perform vehicle make and model recognition for video surveillance using a database of partial license plate and vehicle visual description data. Adaboost and cascaded classifiers are used to detect the license plate. Given the license plate, visual features are extracted using two feature-based methods (SIFT and shape context matching) and one appearance-based method (Eigencars). The drawbacks of the proposed system are that it does not perform color inference, is relatively slow, and only the license plate recognition stage is done in real-time.

SUMMARY

[0007] Vehicle classification has evolved into a significant subject of study due to its importance in autonomous navigation, traffic analysis, surveillance and security systems, and transportation management. While numerous approaches have been introduced for this purpose, no specific study has been able to provide a robust and complete video-based vehicle classification system based on the rear-side view

where the camera's field of view is directly behind the vehicle. In accordance with an embodiment, a stochastic multi-class vehicle classification system is presented, which classifies a vehicle (given its direct rear-side view) into one of four classes Sedan, Pickup truck, SUV/Minivan, and unknown. A feature pool of low-level tail
5 light and vehicle dimension features is extracted, which feeds a feature selection algorithm to define a low-dimensional feature vector. The feature vector is then processed by a Hybrid Dynamic Bayesian Network (HDBN) to classify each vehicle.

[0008] In accordance with an exemplary embodiment, a method for vehicle
10 classification comprises: performing a Bayesian network analysis for vehicle classifications, which are known, wherein the Bayesian network is defined as a directed acyclic graph $G = (V, E)$ where the nodes (vertices) represent random variables from the domain of interest and the arcs (edges) symbolize the direct dependencies between the random variables.

15 [0009] In accordance with a further exemplary embodiment, a system for classification of vehicles comprises: a camera for capturing images of at least one moving object; and a computer processing unit, which performs the steps as recited herein.

[0010] In accordance with another exemplary embodiment, a computer program
20 product comprising a non-transitory computer usable medium having a computer readable code embodied therein for classification of passenger vehicles and measuring their properties from a rear view video frame, the computer readable program code is configured to execute a process, which performs the steps as recited herein.

25 [0011] The details of one or more embodiments of the disclosure are set forth in the accompanying drawings and the description below. Other features, objects, and advantages will be apparent from the description and drawings, and from the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

30 [0012] The accompanying drawings are included to provide a further understanding of the invention, and are incorporated in and constitute a part of this specification. The drawings illustrate embodiments of the invention and, together

with the description, serve to explain the principles of the invention. In the drawings,

[0013] Figure 1 shows a series of images of a direct rear view of a moving vehicle.

[0014] Figure 2 shows the complete system framework for vehicle classification in video in accordance with an exemplary embodiment.

[0015] Figure 3 shows shadow removal and obtaining a bounding box in accordance with an exemplary embodiment.

[0016] Figure 4 shows a series of images for tail light detection in accordance with an exemplary embodiment.

[0017] Figure 5 shows a feature selection subset correct classification rate (CCR) plot.

[0018] Figures 6(a) and 6(b) show Bayesian Network Structures in accordance with an exemplary embodiment.

[0019] Figure 7 shows a Dynamic Bayesian Network (DBN) structure for time slices t_i , wherein $i = 1, 2, 3$ in accordance with an exemplary embodiment.

[0020] Figure 8 shows a series of images of examples of unknown vehicles.

[0021] Figure 9 shows a pattern feature matrix for different vehicles in accordance with an exemplary embodiment.

[0022] Figure 10 shows a series of images of examples of misclassification in accordance with an exemplary embodiment.

[0023] Figure 11 shows a performance Receiver Operating Characteristic (ROC) plot in accordance with an exemplary embodiment.

DETAILED DESCRIPTION

[0024] In accordance with an exemplary embodiment, a probabilistic classification framework is disclosed, which determines the class of a vehicle given its direct rear view (Figure 1). It can be appreciated that the main classes can include Sedan, Pickup truck, SUV/minivan, and also a class for unknown vehicles. In accordance with an embodiment, SUVs and minivans were combined because the number of available minivans in the dataset was small. However, it would be desirable to distinguish minivans from SUVs after obtaining a reasonable number of minivans for our training dataset.

[0025] In accordance with an exemplary embodiment, the direct rear view is chosen for two main reasons. First, most of the research in this area has focused on the side view, whereas the frontal view and rear view have been less investigated. Secondly, the license plate location is a descriptive feature of a vehicle's class (e.g., usually pickup truck's license plate is mounted on the bumper) but not all states require a front license plate (e.g., 19 states in the USA require only the rear license plate).

[0026] In accordance with an exemplary embodiment, a Hybrid Dynamic Bayesian Network (HDBN) classifier with multiple time slices corresponding to multiple video frames to increase classification accuracy is disclosed. In accordance with an exemplary embodiment, the need for high resolution images/videos is eliminated by using simple low-level features (e.g., height, width, and angle), which are also computationally inexpensive, thus, the proposed method is capable of running in real-time.

15

[0027] **Technical Approach**

[0028] The complete proposed system pipeline is shown in Figure 2, which includes data collection, feature extraction, feature selection, and classification. Each of the components are explained in the following sections.

20

[0029] **Feature Extraction**

[0030] In accordance with an embodiment, three main types of features are extracted from the images; tail lights, license plate, and rear dimensions. The tail light features include separately for each tail light, a width, distance from the license plate, and an angle between tail light and license plate. The license plate location and size is used as a reference to enable comparison and help normalize tail light properties and vehicle size values. The feature extraction component consists of three subcomponents: vehicle detection, license plate extraction, and tail light extraction.

30

[0031] Vehicle Detection

[0032] In accordance with an embodiment, a Gaussian mixture model approach is used for moving object detection. The Gaussian distributions are used to determine if a pixel is more likely to belong to the background model or not. An AND
5 approach is used which determines a pixel as background only if it falls within three standard deviations for all the components in all three R, G, and B color channels (i.e., red, green, and blue color channels). The detected moving object is validated by using a simple frame differencing approach and cross checking the masks from both methods.

10 **[0033]** The resulting mask may include some shadow and erroneous pixels. The shadow is removed by finding the vertical axis of symmetry using an accelerated version of Loy's symmetry and readjusting the bounding box containing the mask with respect to the axis of symmetry. This is preferably done by measuring the distance between each point on both vertical sides of the bounding box and the axis
15 symmetry and moving the vertical side that is farther away closer to the axis of symmetry such that each side has the same distance from it. Figure 3 shows results from multiple steps of this approach. The aforementioned shadow removal method fails if the shadow is behind the vehicle. In such cases, the shadow is preferably removed using the approach introduced by Nadimi et al., which does not rely on the
20 common geometrical assumptions such as camera location, object geometry, and ground surface geometry. Given the vehicle rear mask, the height and width of the bounding box and area of the mask are measured.

[0034] License Plate Extraction

25 **[0035]** The license plate corner coordinates are input into the algorithm. It can be appreciated that there are a number of algorithms for license plate extraction. In accordance with an exemplary embodiment, the corners of the license plate are manually extracted. To have more realistic license plate location coordinates, Gaussian noise with constant mean 0 and variance 0.2 time width is added to the
30 license plate width measurement.

[0036] Tail Light Extraction

[0037] For tail light detection, the regions of the image where red color pixels are dominant are located. The redness of each image pixel can be computed by fusing two methods. In the first approach, the image is converted to HSV (hue, saturation, and value) color space and then pixels are classified into three main color groups red, green, and blue. The second method proposed by Gao et. al defines the red

level of each pixel as $r_i = \frac{2r_i}{G_i + B_i}$ in RGB (red, green, blue) color space. A

bounding box surrounding each tail light is generated by combining results from both methods and checking if the regions with high redness can be a tail light (e.g., are symmetric, are close to the edges of the vehicle). Figure 4 presents results of the two methods and the combined results as two bounding boxes. It can be appreciated that both these methods fail, if the vehicle body color is red itself. To overcome this, in accordance with another exemplary embodiment, the vehicle color is estimated using a HSV color space histogram analysis approach, which determines if the vehicle is red or not. If a red vehicle is detected, the tail light detection component is enhanced by adding an extra level of post-processing, which can include Otsu's thresholding, color segmentation, removing large and small regions, and symmetry analysis. After the tail lights are detected, the width, centroid, and distance and angle with the license plate are separately computed for both left and right tail lights

20

[0038] Feature Pool

[0039] As the result of the feature extraction component, the following 11 features are extracted from each image frame (all distances are normalized with respect to the license plate width):

- 25 1. perpendicular distance from license plate centroid to a line connecting two tail light centroids;
2. right tail light width;
3. left tail light width;
4. right tail light-license plate angle;
- 30 5. left tail light-license plate angle;
6. right tail light-license plate distance;

7. left tail light-license plate distance;
8. bounding box width;
9. bounding box height;
10. license plate distance to bounding box bottom side; and
- 5 11. vehicle mask area.

[0040] Given a set of features Y , feature selection determines a subset X which optimizes an evaluation criterion J . Feature selection is performed for various reasons including improving classification accuracy, shortening computational time, reducing measurements costs, and relieving the curse of dimensionality. In accordance with an exemplary embodiment, a Sequential Floating Forward Selection (SFFS) was chosen, which returns a single suboptimal solution. SFFS starts from an empty set and adds the most significant features (e.g., features that increase accuracy the most). The SFFS provides a kind of back tracking by removing the least significant feature during the third step, conditional exclusion. A stopping condition is required to halt the SFFS algorithm, therefore, in accordance with an exemplary embodiment, the number of feature selection iterative steps is limited to 2^{n-1} (where n is the number of features) and a correct classification rate (CCR) threshold of $b\%$ where b is greater than the CCR of the case when all features are used is defined. In other words, the algorithm stops when either the CCR is greater than $b\%$, or 2^{n-1} iterations are completed. The pseudo code for SFFS is shown below, wherein k is the number of features already selected.

1. Initialization: $k = 0; X_0 = \{\emptyset\}$
2. Inclusion: add the most significant feature

$$x_{k+1} = \arg \max_{x \in (Y - X_k)} [J(X_k + x)]$$

$$X_{k+1} = X_k + x_{k+1}; \text{ repeat step 2 if } k < 2$$
- 25 3. Conditional Exclusion: find the least significant feature and remove (if not last added)

$$x_r = \arg \max_{x \in X_k} [J(X_k - x)]$$
 if $x_r = x_{k+1}$ then $k = k + 1$; Go to step 1

 else $X'_k = X_{k-1} - x_r$
- 30

4. Continuation of Conditional Exclusion
 - $x_s = \arg \max_{x \in X_k} [J(X'_k - x)]$
 - if $J(X'_k - x_s) \leq J(X_{k-1})$ then
 - $X_k = X'_k$; Go to step 2
 - 5 else $X'_{k-1} = X'_k - x_s$; $k = k + 1$
5. Stopping Condition Check
 - if *halt_condition* = true then STOP
 - else Go to step 4

[0041] Figure 5 shows the correct classification rate plot with feature selection steps as the x-axis and correct classification rate as the y-axis. The plot peaks at x = 10 5 and the algorithm returns features 1, 4, 6, 10, and 11 as the suboptimal solution.

[0042] **Classification**

[0043] **1) Known or Unknown**

15 [0044] The classification component consists of a two stage approach. Initially, the vehicle feature vector is classified as known or unknown. To do such, the Gaussian distribution parameters of the distance to the nearest neighbor for all vehicles in the training dataset are estimated. To determine if a vehicle test case is known or unknown first the distance to its nearest neighbor is computed. Then 20 following the empirical rule if the distance does not lie within 4 standard deviations of the mean ($\mu \pm 4\sigma$) it is classified as unknown. If the vehicle is classified as known it is a candidate for the second stage of classification.

[0045] **2) DBNs for Classification**

[0046] In accordance with an exemplary embodiment, the use of a Dynamic 25 Bayesian Networks (DBNs) for vehicle classification in video is proposed. Bayesian networks offer a very effective way to represent and factor joint probability distributions in a graphical manner, which makes them suitable for classification purposes. A Bayesian network is defined as a directed acyclic graph $G = (V, E)$ where the nodes (vertices) represent random variables from the domain of 30 interest and the arcs (edges) symbolize the direct dependencies between the random

variables. For a Bayesian network with n nodes X_1, X_2, \dots, X_n the full joint distribution is defined as:

$$p(x_1, x_2, \dots, x_n) = p(x_1) \times p(x_2 | x_1) \times \dots \times p(x_n | x_1, x_2, \dots, x_{i-1}) = \prod_{i=1}^n p(x_i | x_1, \dots, x_{i-1})$$

5 but a node in a Bayesian network is only conditional on its parent's values so

$$p(x_1, x_2, \dots, x_n) = \prod_{i=1}^n p(x_i | \text{parents}(X_i))$$

where $p(x_1, x_2, \dots, x_n)$ is an abbreviation for $p(X_1 = x_1 \wedge \dots \wedge X_n = x_n)$. In other words, a Bayesian network models a probability distribution if each variable is conditionally independent of all its non-descendants in the graph given the value of
 10 its parents.

[0047] In accordance with an embodiment, the accuracy of the model is based on the structure of the Bayesian network. For example, learning the best structure/topology for a Bayesian network takes exponential time because the numbers of possible structures for a set of given nodes is super-exponential in the
 15 number of nodes. To avoid performing exhaustive search, a K2 algorithm (Cooper and Herskovits, 1992) may be used to determine a sub-optimal structure. K2 is a greedy algorithm that incrementally add parents to a node according to a score function. In accordance with an exemplary embodiment, a BIC (Bayesian Information Criterion) function may be used as the scoring function. Figure 6(a)
 20 illustrates the resulting Bayesian network structure. In accordance with another exemplary embodiment, a manually structured network (Figure 6(b)) is defined and the two structures are compared. The details for each node are as following:

1. C: vehicle class, discrete hidden node, size=3
2. LP: license plate, continuous observed node, size=2
- 25 3. LTL: left tail light, continuous observed node, size=3
4. RTL: right tail light, continuous observed node, size=3
5. RD: rear dimensions, continuous observed node, size=3

[0048] For continuous nodes the size indicates the number of features each node is representing, and for the discrete node C it denotes the number of classes. RTL and

LTL are continuous nodes and each contain the normalized width, angle with the license plate, and normalized Euclidean distance with the license plate centroid. LP is a continuous node with distance to the bounding box bottom side and perpendicular distance to the line connecting the two tail light centroids as its features. RD is a continuous node with bounding box width and height, and vehicle mask area as its features. For each continuous node of size n , a multivariate Gaussian conditional probability distribution (CPD) is defined, where each feature of each continuous node has $\mu = [\mu_1, \dots, \mu_n]^T$ and Σ as an $n \times n$ symmetric, positive definite covariance matrix. The discrete node C has a corresponding conditional probability table (CPT) assigned to it which defines the probabilities $P(C = \text{sedan})$, $P(C = \text{pickup})$, and $P(C = \text{SUV or minivan})$.

[0049] Adding a temporal dimension to a standard Bayesian network creates a DBN. The time dimension is explicit, discrete, and helps model a probability distribution over a time-invariant process. In accordance with an exemplary embodiment, a DBN is created by replicating a Bayesian network with time-dependent random variables over T time slices. A new set of arcs defining the transition model is also used to determine how various random variables are related between time slices.

[0050] In accordance with an exemplary embodiment, a video based classifier is modeled by extending the aforementioned Bayesian network (Figure 6(b)) to a DBN. The DBN structure is defined as following:

1. for each time slice $t, i=1, 2, \dots, 5$ the DBN structure is similar to the Bayesian network structure given in Figure 6(b).
2. each feature X_i^t is the parent of X_i^{t+1} .
3. C^t is the parent of C^{t+1} .
4. all intra slice dependencies (arcs) also hold as inter time slices except for arcs from time slice t hidden nodes to time slice $t + 1$ observed nodes.

[0051] Figure 7 demonstrates the DBN structure for 3 time slices. It can be appreciated that such a network is identified as a Hybrid Dynamic Bayesian Network (HDBN) because it consists of discrete and continuous nodes. Training the HDBN or in other words learning the parameters of the HDBN is required before

classification is performed. Therefore, the probability distribution for each node given its parents should be determined. For time slice t_1 this includes $p(\text{LTL}|C)$, $p(\text{RTL}|C)$, $p(\text{RD}|C)$, $p(\text{LP}|C, \text{LTL}, \text{RTL})$, and $p(C)$. For time slices $t_{i, i=2, \dots, 5}$ it includes:

$$\begin{aligned}
 & p(\text{LTL}^t | C^t, \text{LTL}^{t-1}), p(\text{RTL}^t | C^t, \text{RTL}^{t-1}) \\
 & p(C^t | C^{t-1}), p(\text{RD}^t | C^t, \text{RD}^{t-1}) \\
 & p(\text{LP}^t | C^t, \text{LTL}^t, \text{RTL}^t, \text{LP}^{t-1}, \text{LTL}^{t-1}, \text{RTL}^{t-1})
 \end{aligned}$$

[0052] For example, to determine $p(\text{LTL}^t | C^t, \text{LTL}^{t-1})$ three distributions with different parameters, one for each value of C^t , are required. Hence, $p(\text{LTL}^t | \text{LTL}^{t-1}, C^t = \text{sedan})$, $p(\text{LTL}^t | \text{LTL}^{t-1}, C^t = \text{pickup})$, and $p(\text{LTL}^t | \text{LTL}^{t-1}, C^t = \text{SUV or Minivan})$ are estimated, and $p(\text{LTL}^t | \text{LTL}^{t-1})$ is derived by summing over all the C^t cases.

[0053] The next step is inference where a probability distribution over the set of vehicle classes is assigned to the feature vector representing a vehicle. In other words inference provides $p(C^t | f^{(1:t)})$ where $f^{(1:t)}$ refers to all features from time slice t_1 to t_5 .

[0054] **Experiments**

[0055] **Data Collection**

[0056] In accordance with an exemplary embodiment, video data of passing vehicles was captured using a Sony HDR-SR12 Video Camera. The videos are taken in the early afternoon with sunny and partly cloudy conditions. Lossless compressed PNG image files were extracted from the original HD MPEG4 AVC/H.264 video format, and then down sampled from 1440 x 1080 to 320 x 240 using bicubic interpolation. Downsampling is performed to reduce the computation time. All image frames were manually labeled with the vehicle class to provide the ground-truth for evaluation purposes.

[0057] The dataset consists of 100 sedans, 27 pickup trucks, and 42 SUV/minivans. Before extracting the features and generating the feature pool, it is important to determine the number of frames required for classification. Classification accuracy was recorded for different number of frames. The maximum accuracy is achieved when 5 frames are used. Note that these frames are not

successive. In accordance with an exemplary embodiment, $\Delta t=2$ was used, which means leaving out two frames between candidate frames. This value is directly related to the speed of the vehicle and the overall time the vehicle is visible in the camera's field of view.

5 [0058] To evaluate how well the algorithm performs in the case of an unknown vehicle, 8 unknown vehicles were collected, which were not part of the training dataset. Figure 8 shows two examples of unknown vehicles.

[0059] Figure 9 shows the corresponding pattern-feature matrix. The y-axis represents the extracted features and the x-axis symbolizes all 845 (169 vehicles
10 times 5 frames) feature vectors. For presentation purposes each row has been normalized by dividing by the maximum value of the same feature.

[0060] **Feature Selection Evaluation**

[0061] Table I shows classification evaluation metrics both when (a) using the
15 entire feature set, and (b) using a suboptimal subset. Results show that using the subset of the features generated by SFFS decreases the accuracy and precision by approximately 1%. Feature selection also decreases the average testing time per frame from 0.05 to 0.03 seconds.

20

TABLE 1
Feature Selection Results

FS Method↓	Selected features	Prec-ision	FA Rate	CCR	Testing time(s)
(a) None	all	95.68	0.02	97.63	0.05
(b) SFFS	1, 4, 6, 10, 11	94.23	0.03	96.68	0.03

[0062] **Classification Results**

25 [0063] In accordance with an exemplary embodiment, a Bayes Net Toolbox (BNT), an open source Matlab package, was used for defining the DBN structure, parameter learning, and computing the marginal distribution on the class node. In accordance with an exemplary embodiment, the proposed classification system was tested on a dataset consisting of 169 known and 8 unknown vehicles. A stratified k-

fold cross-validation with $k=10$ was used to evaluate the approach. The resulting confusion matrix is shown in Table II. As shown, all sedans are correctly classified except for the one that is misclassified as a pickup truck (Figure 10(a)). Figure 10(b) shows an SUV misclassified as a pickup truck. A closer look at the data and pattern-feature matrix shows great similarity for both these cases with the pickup class due to the license plate location and rear tail light width.

TABLE II
Confusion Matrix

Pred. Class→ True Class↓	Unknown	Sedan	Pickup	SUV/ Minivan	Total
Unknown	8	0	0	0	8
Sedan	0	99	1	0	100
Pickup	0	0	27	0	27
SUV/Minivan	0	3	2	37	42

[0064] Structure Learning Evaluation

[0065] Table III presents the classification evaluation metrics for the two structures given in Figure 6(a) and Figure 6(b). The results show that learning the structure using K2 decreases the classification accuracy and precision. This is because the K2 search algorithm requires a known linear ordering of nodes prior to model selection. One way to overcome this is to determine the ordering of nodes prior to performing K2. Determining the required ordering using a dynamic programming approach takes $O(n^2 2^n)$ time and $O(n 2^n)$ space where n is the number of nodes. The linear order determines the possible parent candidates for each node in a way that the BN is guaranteed to be an acyclic graph.

TABLE III
Structure Learning Evaluation

Structure Learning	Precision	FA Rate	CCR
K2&BIC (Fig. 6(a))	93.68	0.04	96.06
Manual (Fig. 6(b))	95.68	0.02	97.63

[0066] Comparison with Other Methods

[0067] The results were compared with 3 well-known classifiers: k-nearest neighbor (kNN), linear discriminant analysis (LDA), and support vector machines (SVM). Tables IV, V, and VI show classification accuracy, false positive ratio (false alarm), and precision respectively. The class "unknown" is not included in computing the results for Tables IV, V, and VI.

**TABLE IV
CCR COMPARISON**

10

Classifier→ Vehicle Class↓	kNN	LDA	SVM	HDBN
Sedan	88.25	94.67	96.44	97.63
Pickup	95.12	94.67	96.44	98.22
SUV/Minivan	90.90	92.89	92.30	97.04
Overall	91.42	94.07	95.06	97.63

**TABLE V
FALSE ALARM PERCENTAGES COMPARISON**

15

Classifier→ Vehicle Class↓	kNN	LDA	SVM	HBN
Sedan	0.17	0.07	0.06	0.04
Pickup	0.04	0.05	0.03	0.02
SUV/Minivan	0.04	0.02	0.04	0
Overall	0.09	0.05	0.04	0.02

**TABLE VI
PRECISION PERCENTAGES COMPARISON**

20

Classifier→ Vehicle Class↓	kNN	LDA	SVM	HBN
Sedan	88.46	95.05	96.07	97.05
Pickup	80.64	78.13	86.20	90.00
SUV/Minivan	85.29	91.67	87.17	100
Overall	84.80	88.28	89.81	95.68

[0068] Figure 11 shows the Receiver Operating Characteristic (ROC) curves for all the four classifiers. Although the ROC curves are similar, it is clear that HDBN outperforms SVM, LDA, and KNN.

5 [0069] **Conclusion**

[0070] In accordance with an exemplary embodiment, a Dynamic Bayesian Network for vehicle classification using multiple video frames in a DBN structure can outperform well known classifiers such as kNN, LDA, and SVM. Experimental results showed that obtaining high classification accuracy does not always require
10 high level features and simple features (e.g., normalized distance and angle) may also provide such results making it possible to perform real-time classification.

[0071] In accordance with another exemplary embodiment, a computer program product comprising a non-transitory computer usable medium having a computer readable code embodied therein for classification of passenger vehicles and
15 measuring their properties from a rear view video frame, the computer readable program code is configured to execute a process, which performs the steps as recited herein.

[0072] The computer usable medium, of course, may be a magnetic recording medium, a magneto-optic recording medium, or any other recording medium which
20 will be developed in future, all of which can be considered applicable to the present invention in all the same way. Duplicates of such medium including primary and secondary duplicate products and others are considered equivalent to the above medium without doubt. Furthermore, even if an embodiment of the present invention is a combination of software and hardware, it does not deviate from the
25 concept of the invention at all. The present invention may be implemented such that its software part has been written onto a recording medium in advance and will be read as required in operation.

[0073] It will be understood that the foregoing description is of the preferred embodiments, and is, therefore, merely representative of the article and methods of
30 manufacturing the same. It can be appreciated that many variations and modifications of the different embodiments in light of the above teachings will be readily apparent to those skilled in the art. Accordingly, the exemplary

embodiments, as well as alternative embodiments, may be made without departing from the spirit and scope of the articles and methods as set forth in the attached claims.

5

WHAT IS CLAIMED IS:

1. A method for vehicle classification comprising:
 performing a Bayesian network analysis for vehicle classifications, which are
 5 known, wherein the Bayesian network is defined as a directed acyclic graph $G = (V, E)$, where the nodes represent random variables from the domain of interest and the arcs symbolize the direct dependencies between the random variables.

2. The method of Claim 1, wherein for a Bayesian network with n nodes
 10 X_1, X_2, \dots, X_n , a full joint distribution is defined as:

$$p(x_1, x_2, \dots, x_n) = p(x_1) \times p(x_2 | x_1) \times \dots \times p(x_n | x_1, x_2, \dots, x_{i-1}) = \prod_{i=1}^n p(x_i | x_1, \dots, x_{i-1})$$

but a node in a Bayesian network is only conditional on its parent's values so

$$p(x_1, x_2, \dots, x_n) = \prod_{i=1}^n p(x_i | \text{parents}(X_i))$$

15 where $p(x_1, x_2, \dots, x_n)$ is an abbreviation for $p(X_1 = x_1 \wedge \dots \wedge X_n = x_n)$.

3. The method of Claim 2, further comprising using a K2 algorithm to determine a sub-optimal structure.

20 4. The method of Claim 3, further comprising generating a manually structured network and comparing the two structures.

5. The method of Claim 4, further comprising adding a temporal dimension to the Bayesian network to create a Dynamic Bayesian Network (DBN structure).
 25

6. The method of Claim 5, further comprising replicating the Bayesian network with time-dependent random variables over T time slices, and wherein a

new set of arcs defining the transition model is used to determine how various random variables are related between time slices.

7. The method of Claim 6, wherein the DBN structure is defined as
5 following:

for each time slice $t_i, i=1, 2, \dots, 5$;

each feature X'_i is the parent of $X'_i{}^{t+1}$;

C^t is the parent of C^{t+1} ; and

all intra slice dependencies also hold as inter time slices except for arcs from
10 time slice t hidden nodes to time slice $t+1$ observed nodes.

8. The method of Claim 7, further comprising generating a Hybrid
Dynamic Bayesian Network (HDBN), which consists of discrete and continuous
nodes, and wherein the parameters of the HDBN is required before classification is
15 performed.

9. The method of Claim 8, wherein the classifications include Sedan,
Pickup, and SUV or Minivan.

10. The method of Claim 1, wherein the complete system framework
20 includes data collection, feature extraction, feature selection, and classification.

11. The method of Claim 10, wherein the features extracted are from
images of a rear view of a vehicle.

25

12. The method of Claim 11, wherein the tail light features include
separately for each tail light a width, a distance from the license plate, and an angle
between the tail light and the license plate.

13. The method of Claim 12, wherein the license plate location and a size thereof is used as a reference to enable comparison and normalize the tail light properties and the vehicle size values.

5 14. The method of Claim 1, further comprising detecting three subcomponents, wherein the three subcomponents are vehicle detection, license plate extraction, and tail light extraction.

10 15. The method of Claim 14, further comprising using a Gaussian mixture model approach for detection of a moving object.

15 16. The method of Claim 15, wherein the Gaussian mixture model comprises Gaussian distributions to determine if a pixel is more likely to belong to a background model or not, and an AND approach, which determines a pixel as background only if the pixel falls within three standard deviations for all the components in all three R, G, and B color channels.

20 17. The method of Claim 16, wherein detecting moving objects is validated by using a simple frame differencing approach.

25 18. The method of Claim 17, wherein shadows and erroneous pixels are removed by finding a vertical axis of symmetry using an accelerated version of Loy's symmetry and readjusting the bounding box containing a mask with respect to an axis of symmetry.

30 19. The method of Claim 18, wherein if the shadow is behind the vehicle, removing the shadow using geometrical assumptions such as camera location, object geometry, and ground surface geometry, and wherein given the vehicle rear mask, the height and width of the bounding box, and area of the mask are measured.

20. The method of Claim 19, further comprising inputting the license plate corner coordinates into an algorithm for license plate extraction, and adding a

Gaussian noise with constant mean 0 and variance 0.2 time width to the license plate width measurement.

21. The method of Claim 20, further comprising performing a tail light
5 extraction by fusing two methods, a first method wherein an image is converted to hue, saturation and value (HSV) color space and classifying pixels into three main color groups red, green, and blue, and a second method, which defines the red level of each pixel as $r_i = \frac{2r_i}{G_i + B_i}$ in red, green and blue (RGB) color space.

10 22. The method of Claim 21, further comprising generating a bounding box surrounding each tail light by combining results from both methods and checking if the regions with high redness can be a tail light.

23. The method of Claim 22, wherein if the vehicle body color is red
15 itself, estimating a vehicle color using a hue, saturation and value (HSV) color space histogram analysis approach, which determines if the vehicle is red or not, and if a red vehicle is detected, the tail light detection component is enhanced by adding an extra level of post-processing which includes thresholding, color segmentation, removing large and small regions, and symmetry analysis, and after the tail lights are
20 detected, separately computing width, centroid, and distance and angle with the license plate for both left and right tail lights.

24. The method of Claim 23, further comprising extracting one or more
of the following features for each image frame, and normalizing each distance with
25 respect to the license plate width, the features comprising: perpendicular distance from license plate centroid to a line connecting two tail light centroids, right tail light width, left tail light width, right tail light license plate angle, left tail light license plate angle, right tail light license plate distance, left tail light license plate distance, bounding box width, bounding box height, license plate distance to
30 bounding box bottom side, and/or vehicle mask area.

25. The method of Claim 24, further comprising optimizing an evaluation criterion to improve classification accuracy, shorten computational time, reduce measurements costs, and/or eliminate use of dimensionality.

5 26. The method of Claim 25, further comprising performing a Sequential Floating Forward Selection (SFFS), which is a deterministic statistical pattern recognition (SPR) feature selection method and returning a single suboptimal solution.

10 27. The method of Claim 26, wherein SFFS starts from an empty set and adding features that increase accuracy and removing the least significant features during a conditional exclusion.

15 28. The method of Claim 27, further comprising a stopping condition, which halts the SFFS algorithm to limit the number of feature selection iterative steps to 2^{n-1} , wherein n is the number of features and defining a correct classification rate (CCR) threshold of b%, where b is greater than the CCR of the case when all features are used.

20 29. The method of Claim 28, wherein the classification is a two stage approach, which consists of estimating the Gaussian distribution parameters of the distance to the nearest neighbor for all vehicles in the training dataset and determining if a vehicle test case is known or unknown by initially computing the distance to its nearest neighbor.

25 30. The method of Claim 29, further comprising if the distance does not lie within 4 standard deviations of the mean ($\mu \pm 4\sigma$), classifying it as unknown, and if the vehicle is classified as known it is a candidate for a second stage of classification.

30

31. A stochastic multi-class vehicle classification system, which classifies a vehicle given its direct rear-side view into one of four classes Sedan, Pickup truck, SUV/Minivan, and unknown, the system comprising:
- 5 extracting a feature pool of low-level tail light and vehicle dimension features;
- feeding the features into a selection algorithm to define a low-dimensional feature vector; and
- processing the feature vector with a Hybrid Dynamic Bayesian Network (HDBN) to classify each vehicle.
- 10
32. A system for classification of vehicles comprising:
- a camera for capturing images of at least one moving object; and
- a computer processing unit, which performs the steps of Claims 1-31.
- 15
33. The system of Claim 32, wherein the computer processing unit further comprises a memory arrangement, a processing unit and an optional display unit for displaying data and/or classification of the distinct structures in the class.
- 20
34. The system of Claim 32, wherein the camera captures video images.
- 25
35. A computer program product comprising a non-transitory computer usable medium having a computer readable code embodied therein for classification of passenger vehicles and measuring their properties from a rear view video frame, the computer readable program code is configured to execute a process, which performs the steps of Claims 1-34.

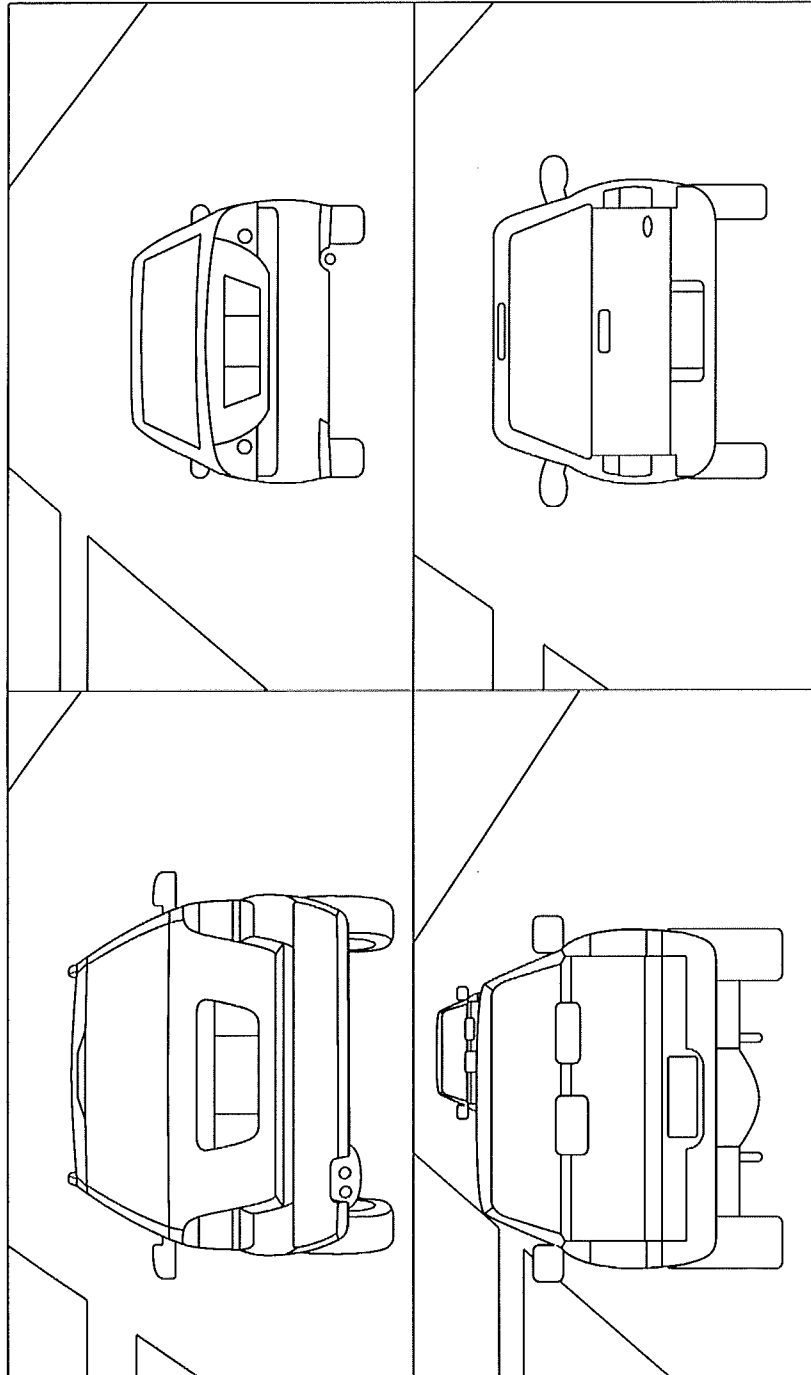


FIG. 1



FIG. 2

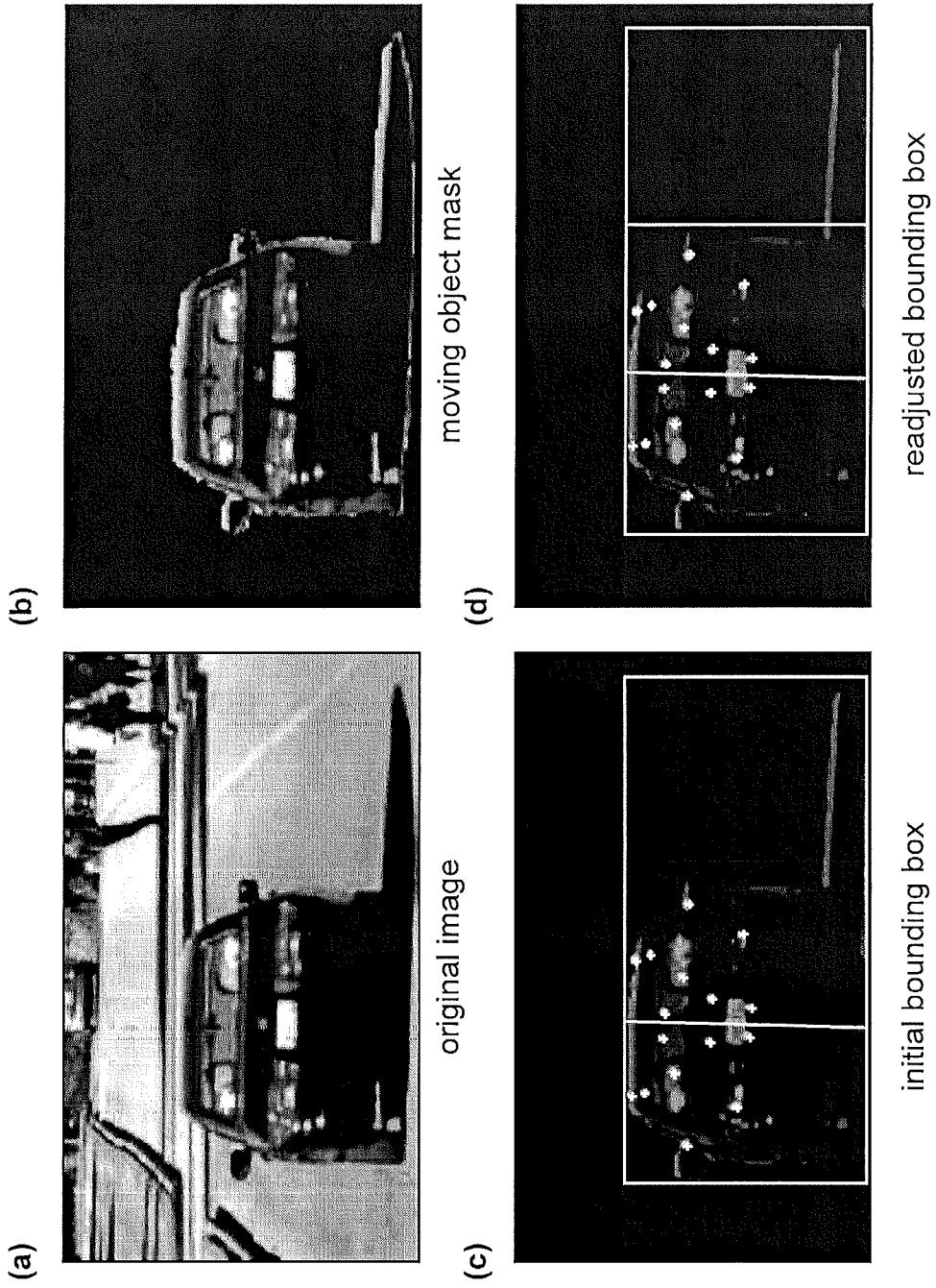


FIG. 3

4 / 11

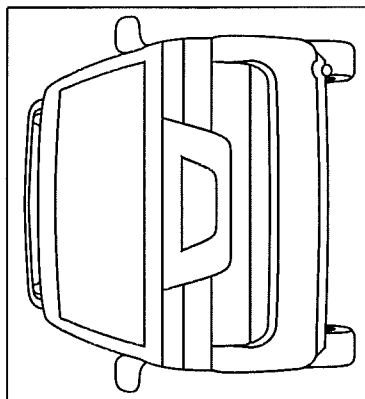
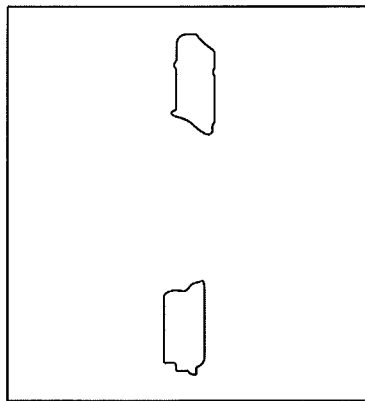
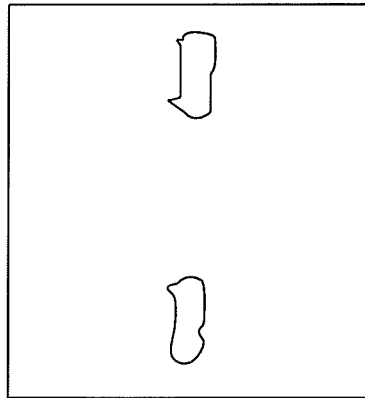
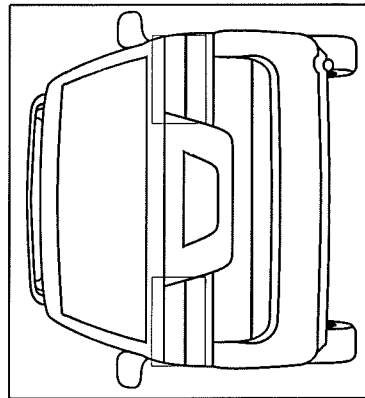


FIG. 4

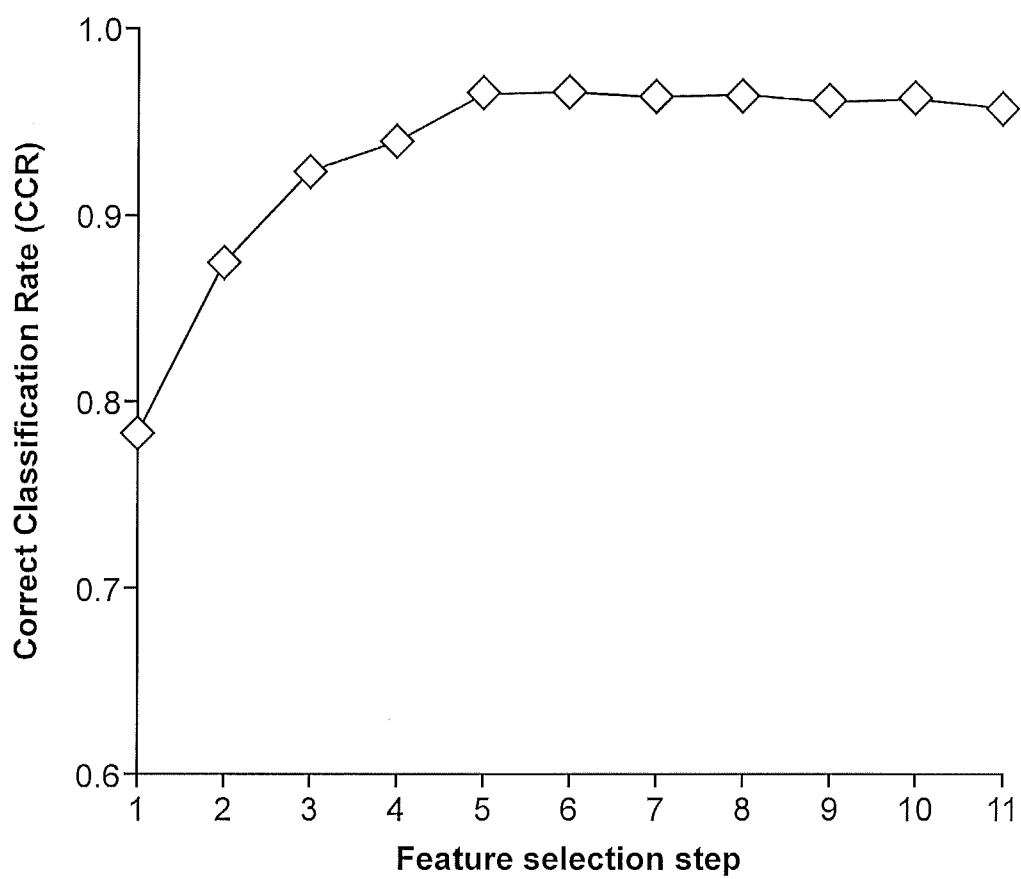
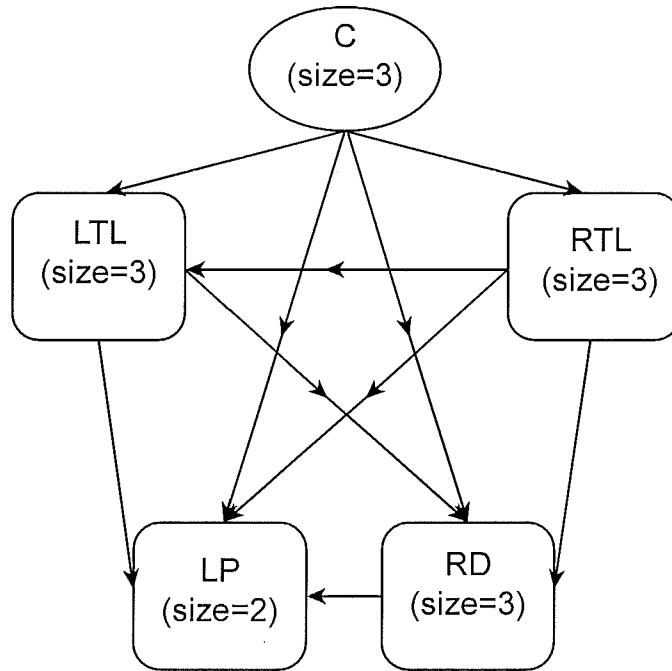


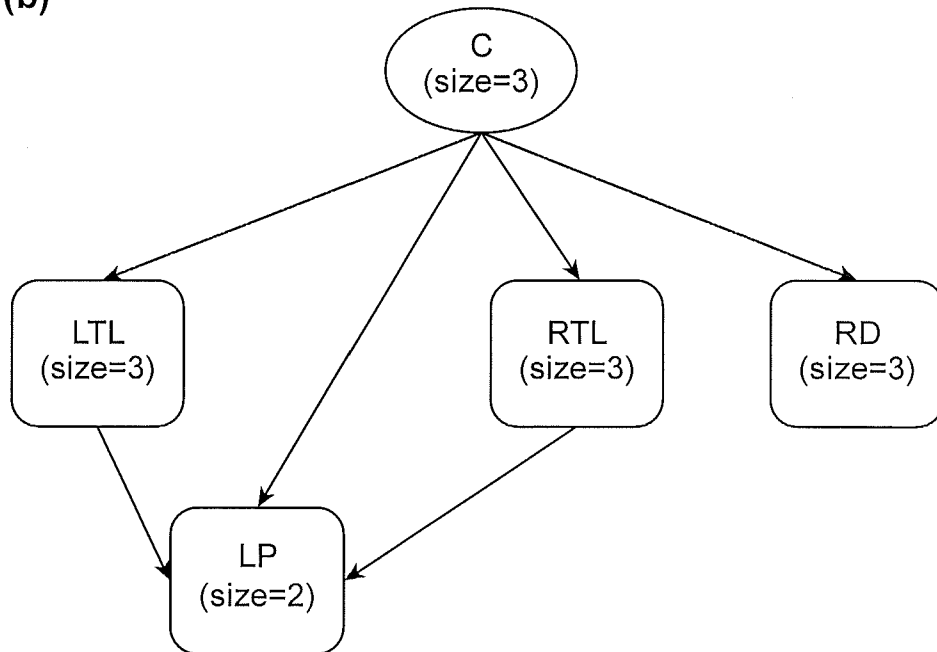
FIG. 5

(a)



K2 generated structure

(b)



Manually structured Bayesian network

FIG. 6

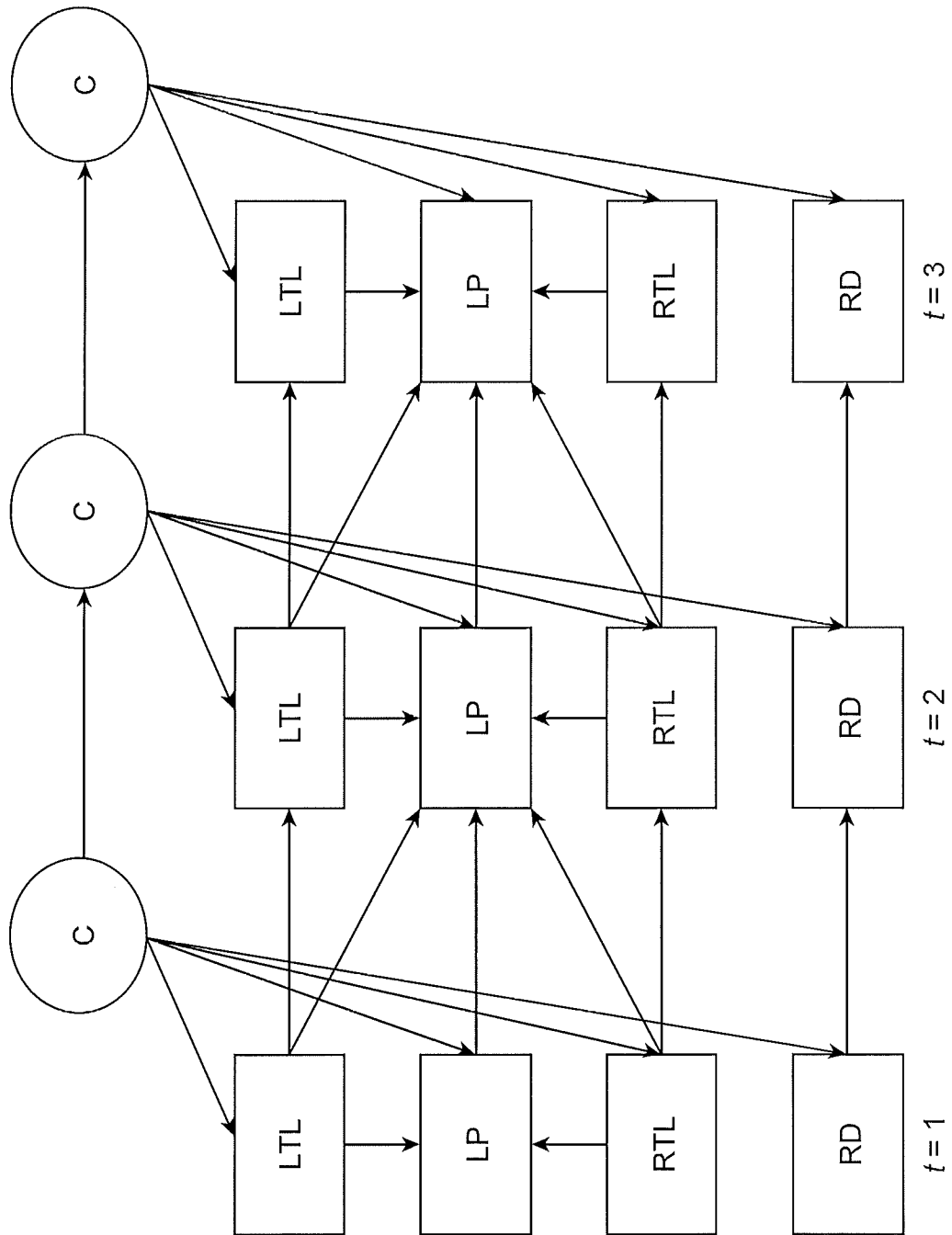


FIG. 7

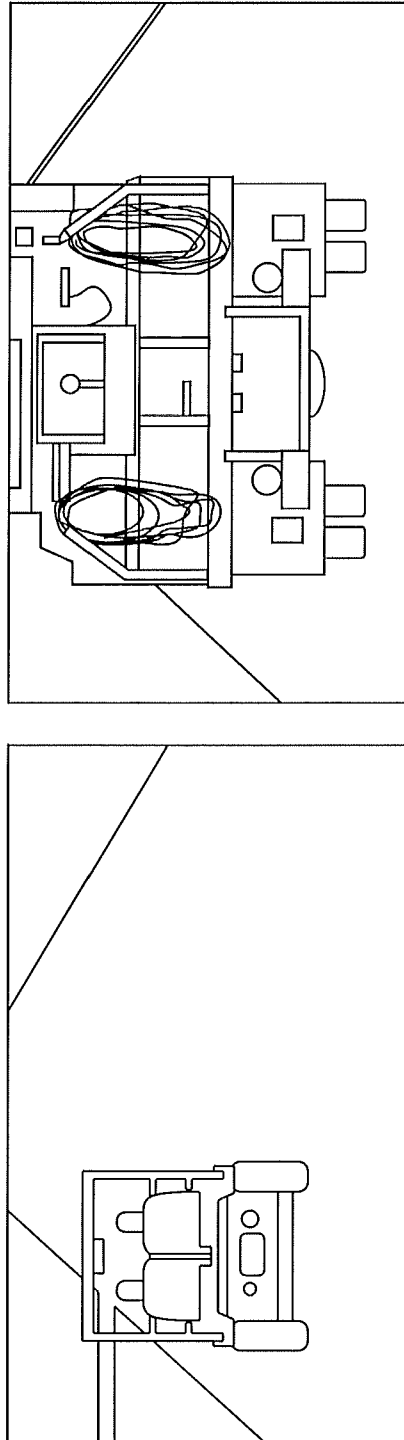


FIG. 8

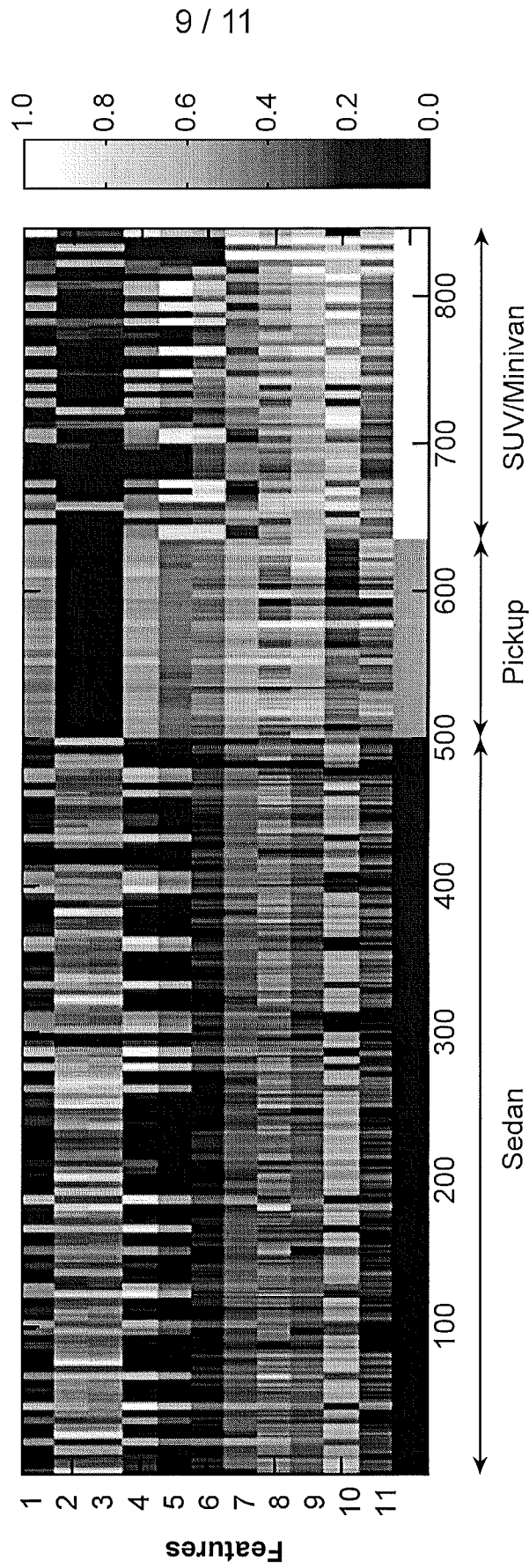


FIG. 9

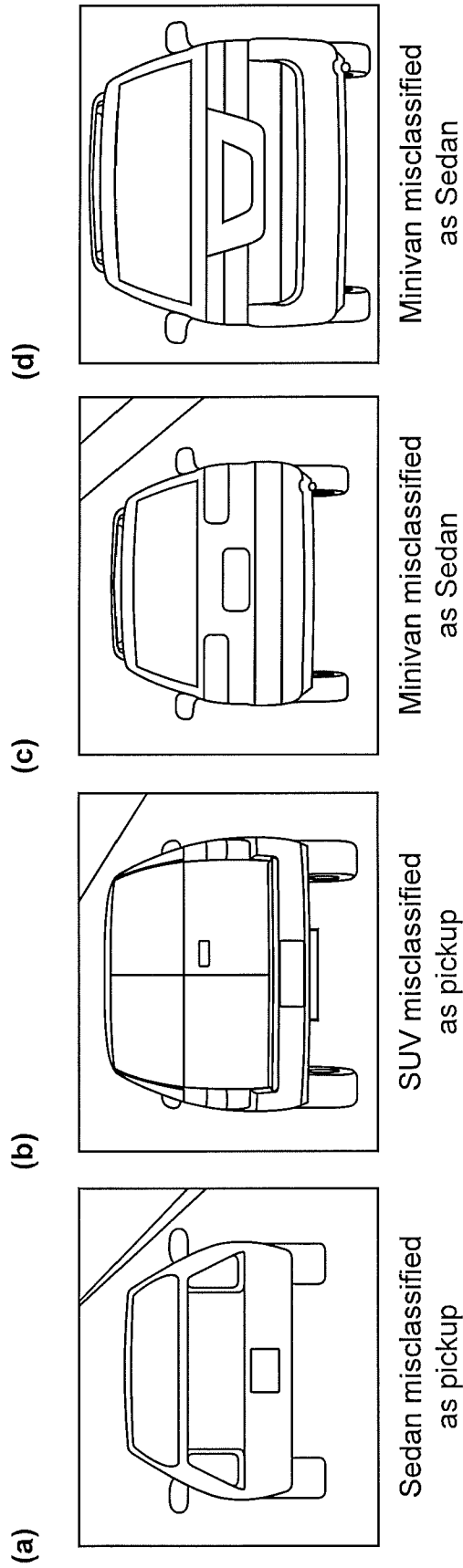


FIG. 10

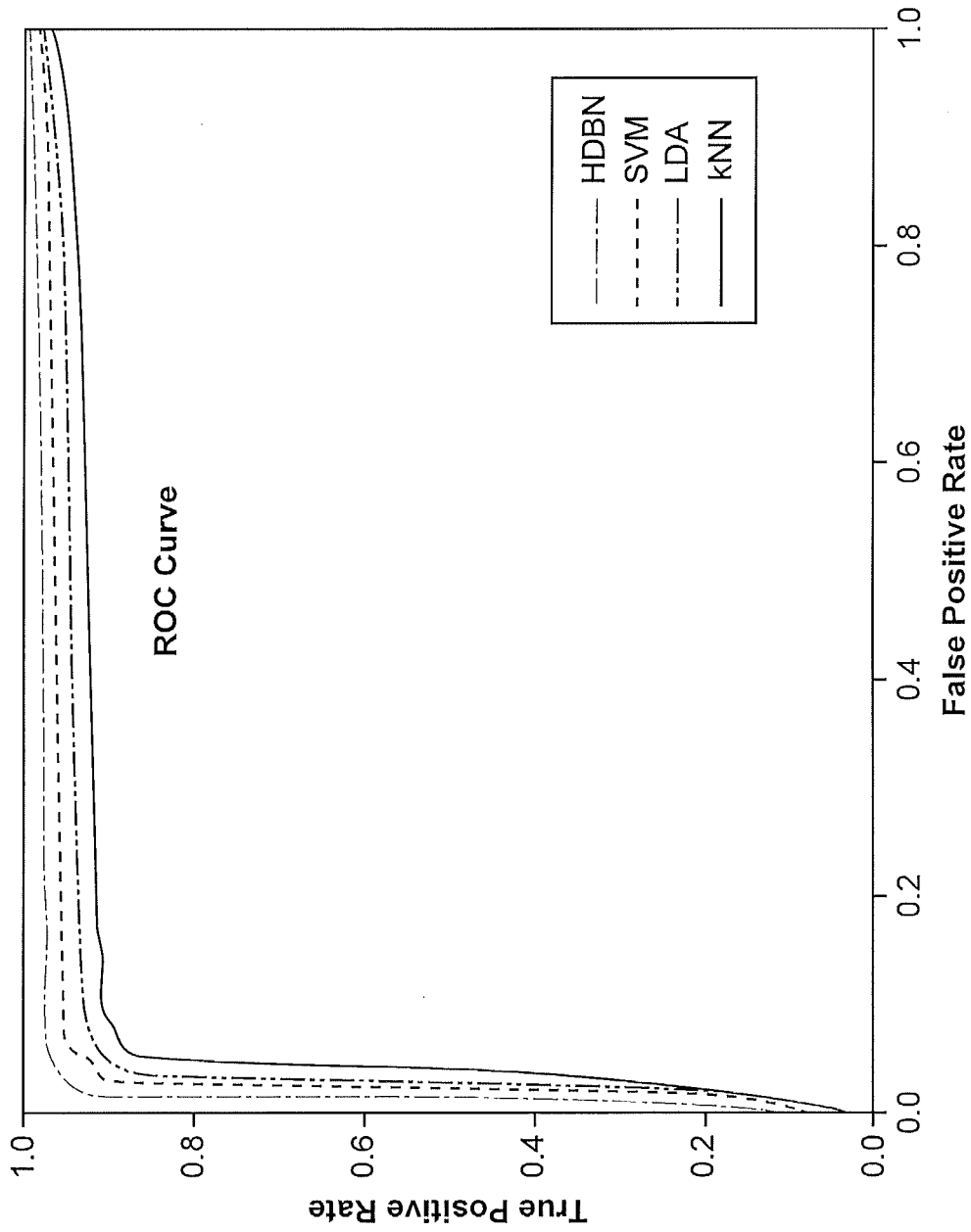


FIG. 11